

①

Optical SMART Board Product Concept

A small, light weight (6 ounce) CMOS or CCD imaging device / lens / drive electronics are mounted on a wire frame in front of a conventional whiteboard. The geometry's, optics and resolutions are set up so that the imaging device captures pixels that are no coarser than $1/20^{\text{th}}$ of an inch square. This means that the largest area that can be captured with a single camera with a resolution of 1024×1024 is about $50'' \times 50''$. Multiple devices can be cascaded to capture long walls of whiteboard. Each image is captured at only 3 bits of color depth (distinguishing up to 7 unique pen colors from white). This means each image is 393K bytes uncompressed. Whiteboard data would generally be susceptible to high compression ratios, probably 100 to 1 without a lot of effort or loss, reducing the average image to 4K bytes.

Initial release: Simple capture

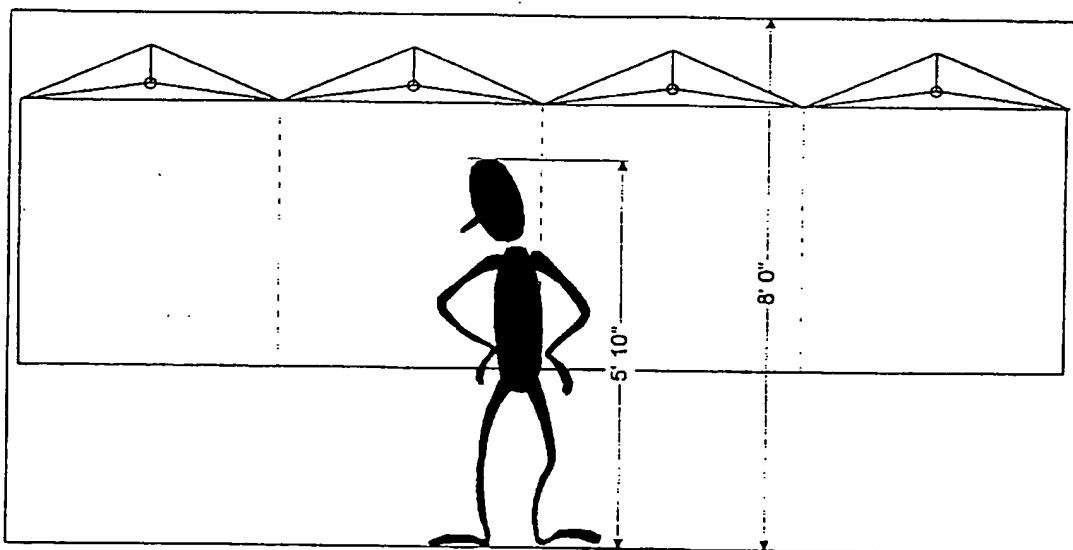
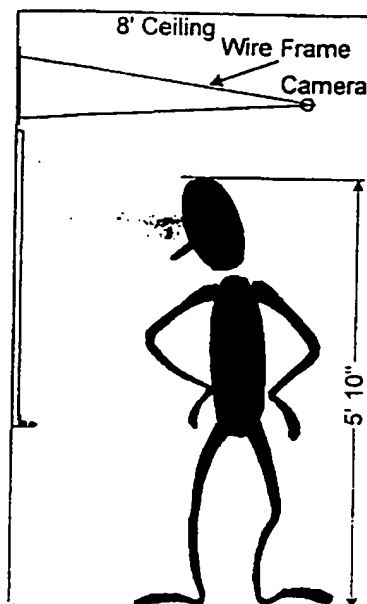
A five button remote control is used by the teacher or meeting scribe to manually trigger a capture of each whiteboard or a capture of all at once. The SMART CameraBook stores each capture as a time sequenced page, and allows pages to be edited, deleted or reordered and of course printed, e-mailed and stored (native or HTML)

Second release: SMART capture

The whiteboard images are continuously monitored by the computer every 2 to 10 seconds or so and only the changes are stored. A "human filter" is required to reject image changes that are due to someone standing (possibly writing) on the board. In this way it is possible to support semi real time data conferencing and replay. When ever a major section of a board is erased, an image of the maximum ink board is archived as a complete whiteboard page with time stamp.

Third release: Stitching

With this release it is no longer necessary to be aware of the 4 foot wide page boundary on the whiteboard. Multiple images are stitched together in software to create a continuous wide image of the whiteboard.



Target COGS: \$300 per Camera and frame